May 2019

Analyzing distances in word embeddings and their relation with seme analysis

Manuel GIJÓN AGUDO [a], Armand VILALTA ARIAS [b] and Dario GARCIA-GASULLA [b,1]

[a] *Universitat Politècnica de Catalunya (UPC)*
[b] *Barcelona Supercomputing Center (BSC)*

Abstract. Word embeddings have recently become a fundamental tool of Natural Language Processing, with application to tasks like machine translation or image annotation. The high-dimensional space defined by these embeddings is typically explored and exploited through distance-based operations. In this paper we work on the problem of finding words related between them in a text embedding. This relationship can be of different kind, we focus in semantic relations like synonymy and antonym. We explore the idea of using the distance between norms instead of, like other authors has done before, the vector that units them. We present different norms, some of them well known in the literature and others no so widely used and also we introduce a new one and its theoretical mathematical framework. We also give an explanation of why them work properly or not and compare their performance on the two most used embeddings, GloVe and Word2Vec.

Keywords. Word embeddings, Embedding space, Distances, Semantic relations, WordNet, High dimensional vector spaces

1. Introduction

In recent years the use of neuronal networks has been increasing due to a significant improvement in the computational power of hardware resources, and to an explosion of digitised information. Deep learning methods (DL) for Natural Language Processing (NLP) are nowadays widely used. Most of these methods relay on a word representation. These representations are typically vectors in a $\mathbb{R}^N$ space (the embedding space). There are different ways to obtain these embeddings, but all of them rely on word concurrences in a corpus of text. Word embeddings capture different types of semantic and syntactic relations between words [1], [9], [10]. These relationships are encoded in the resulting high dimensional space as geometrical relationships.
Text embeddings have many uses, from translation applications [14], text search to multimodal retrieval. Another applications of the regularities presented in these embeddings is also useful to evaluate the quality of the embeddings themselves [8]. One of the most significant ways is based on the relative position of words with similar meaning [8].

[1]Corresponding Authors: Manuel Gijón Agudo, E-mail: manuel.gijon@outlook.es; Armand Vilalta Arias, Jordi Girona 1-3, 08034 Barcelona, Spain. E-mail: armand.vilalta@bsc.es; Dario Garcia-Gasulla, Jordi Girona 1-3, 08034 Barcelona, Spain. E-mail:dario.garcia@bsc.es

Which, combined with the study of the representation of words with opposite meaning (antonyms) gets to the finer level of granularity of seme analysis.
The goals of this paper are:

1. Prove that is possible, using simple statistical test, distinguish between related (synonyms or antonyms) or non related works attending just to the distance that separates them.
2. Check if it is possible to separate between synonyms or antonyms words.
3. Study how well the most used norms in the literature works and compare them with others.
4. Propose a better norm in terms of time compare to the previous alternatives.
5. Compare how well Word2Vec and GloVe embeddings work for this task.

2. Methods

In order to archive our goals, we will be working under the following hypothesis:

1. We hypothesise that the vocabulary include in WordNet (WN) is sufficient to consistently evaluate the quality of the embedding.
2. We hypothesise that distances between pairs vectors have enough relevant information to assess semantic relations. In contrast, others authors have focused on the study of the relationship between two words attending to the characterisation of the vector (direction and relative position in the space) that unites them [9], [10].
3. We hypothesise that distances attending to individual component differences (*e.g.* Canberra, Braycurtis) are more suitable than distances that focus on the average change (*e.g.* euclidean, cosine, correlation).

The method of experimentation we use to test our hypothesis consists of the following steps:

First we gather a set of embeddings pre-trained from online resources. For each of these embeddings, we filter the corresponding vocabulary using the words present in WordNet. Once we define the relation to study, we use Wordnet to create two sets of word pairs based on their fulfilment of the relation. For each of these two sets we sample distances and then we compare them using the Kolmogorov-Smirnov (KS) statistical test. The KS is prefer over other options like Lilliefors, ShapiroWilk or Anderson-Darling because our only goal is to measure how separable by a simple threshold the distributions are, instead of checking the distribution of the data [15]. If the test is significant, we will be able to differentiate between related an unrelated pairs of words using the distance chosen. Repeating this process under different distances we are able to evaluate their usability as a measure of the semantic relationship in high dimensional spaces.

The KS test is the tool that we are going to use to compare our empirical distributions. The statistic measures a distance between the two samples giving us a way to know how different they are.

$$D_{n,m} = \sup_x |F_{1,n}(x) - F_{2,m}(x)| \tag{1}$$

Notice that n and m are respectively the sample sizes of the empirical distributions F_1 and F_2. We reject the null hypothesis (the two samples, $F_{1,n}$ and $F_{2,m}$, follows the same distribution) at a level of confidence α if:

$$D_{n,m} > c(\alpha)\sqrt{\frac{n+m}{nm}}, \quad \text{where } c(\alpha) = \sqrt{-\frac{1}{2}\ln\alpha} \tag{2}$$

2.1. Distances

The most popular distances used in real spaces $\mathbb{R}^N$ are the cosine and the euclidean [13]. Beyond this we wish to explore other distance measures that are not typically evaluated in the literature. The choice of these distances is based on the last of the previous hypothesis, proposing distances than focus on the individual change of the components.

To measure the difference between two vectors, $u = (u_1, u_2, \cdots, u_N)$ and $v = (v_1, v_2, \cdots, v_N)$ where N is the dimension of the embedding space:

- Euclidean distance:

$$d_{\text{euclidean}}(u,v) = \sqrt{\sum_{i=1}^{N}(u_i - v_i)^2} = ||u - v|| \tag{3}$$

- Cosine distance:

$$d_{\text{cosine}}(u,v) = 1 - \frac{u \cdot v}{||u|| \cdot ||v||} \tag{4}$$

  ,where $||\cdot||$ is the euclidean norm of the vector ($||u|| = \sqrt{\sum_{i=1}^{N} u_i^2}$).
- Correlation distance:

$$d_{\text{correlation}}(u,v) = 1 - \frac{(u - \bar{u}) \cdot (v - \bar{v})}{||u - \bar{u}|| \cdot ||v - \bar{v}||} \tag{5}$$

  ,where $\bar{u}$ is the mean of the components ($\bar{u} = \frac{1}{N}\sum_{i=1}^{N} u_i$).
- Canberra distance:

$$d_{canberra}(u,v) = \sum_{i=1}^{N} \frac{|u_i - v_i|}{|u_i| + |v_i|} \tag{6}$$

- Braycurtis distance:

$$d_{\text{braycurtis}}(u,v) = \sum_{i=1}^{N} \frac{|u_i - v_i|}{|u_i + v_i|} \tag{7}$$

Based on the results on the experiments on previous distances, we consider that the key factor is the differences in components. Moreover, we hypothesise that only the change of sing between correspondent components is enough ti properly characterise the embedding. We that in mind, we proposed a new, computationally efficient distance namely Component Sign distance (CS):

$$d_{\mathrm{CS}}(u,v) = \text{Number of coordinates with different sign} \tag{8}$$

Please notice that with this formulation this is not strictly a mathematical distance. See in supplementary materials the proper mathematical definition and justification.

3. Resources

WordNet [2] is one of the most used resources in Natural Language Processing and Representation Learning. The English version of this database includes information over $117,000$ different synsets and their semantic relations(hyponymy, hypernymy, etc.). A synset is a set of terms that represent an unique idea or concept. All the words included in a synset are considered synonymous.

Word2Vec methodology, created by Mikolov et al. [7], [8] and [9] in 2013, automatically creates word embeddings from a corpus of text. Two algorithms are described that produce embeddings. The first one, Continuous Bag-of-Words [5], [6] is trained for predicting a target word given the words around it . On the other hand, the second one, Skip-Gram [5], [6], is to predict the words surrounding a given word.

In our experiments we use the embedding `GoogleNews-vectors-negative300.bin`.[2]. It is a model of dimension 300 and trained with the corpus "Google News dataset".

The GloVe model (Global Vectors for Word Representation) was developed and introduced in 2014 [4] by researchers of Stanford University. In our case, we are using the embeddings `glove.6B.50d.txt`, `glove.6B.100d.txt`, `glove.6B.200d.txt` and `glove.6B.300d.txt` with embedding space dimensions 50, 100, 200 and 300 respectively[3] and trained with the corpus Wikipedia 2014 and Gigaword 5.

In order to facilitate the reproducible of our results we made publicly available the code used at our Github account [4].

4. Experiments

We are going to use the vocabulary and embeddings obtained from previously described resources. For all the GloVe embeddings, we have the same vocabulary set, compose of $400,000$ terms of which, $55,666$ terms are also present in WordNet (13.92%). For the Word2Vec case, we have a total of 3 millions of words of which, only $54,586$ of them are in WordNet (1.82%). For the GloVe embeddings this results in $28,763$ sets of synonyms, that include a total of $30,439$ different words. In the case of Word2Vec embedding, there

[2]This embeding is available in the direction: `https://code.google.com/archive/p/word2vec/`

[3]These embedings are available in the direction: `https://nlp.stanford.edu/projects/glove/`

[4]`https://github.com/MGijon/WER`

are a total of $31,886$ sets of synonyms, with a total of $33,822$ different words. As we can see, the number of synonyms available in both embeddings is quite similar. For the antonyms we follow a similar procedure.

The set of non related words can include or not the words that are not present in WordNet. Both possibilities are studied. The size of the sample of random pairs taken from the non-related ones is limited to $5,000$ for both, filtered and non-filtered vocabularies.

For each of the embeddings considered we do two rounds of experiments: synonyms and antonyms against vocabulary filtered by WordNet, and synonyms or antonyms against all the vocabulary. In each round of experiments, we test each one of the six distances defined previously.

We evaluate this results based on the KS value and the p-values associated. We use $\alpha = 0.05$ as significance value for the test in all the cases.

5. Results

In this section we summarise the main results obtained in our experiments for dimension 300. In the case of the GloVe embeddings (dimensions 50, 100 and 200) the results of these experiments can be found at [16]. For the sake of simplicity, here we just expose the results for dimension 300, the rest of them are in the Github repository[5].

Due to the level of significance and the results, we must reject the null hypothesis in all the cases (*i.e.* the two distributions are not identical).

The tables 1 and 2 contain results of experiments comparing synonyms with random words for the GloVe and Word2Vec embeddings respectively. The results of the experiments for the antonyms in GloVe and Word2Vec embeddings are included in tables 3 and 4. The first surprising result is that the Euclidean distance is the less capable to distinguish between synonyms or antonyms from random pairs of words by a fair margin. In general, Cosine distance is among the best candidates in all experiments except GloVe non-filtered. Given these poor results, this distance will not be considered in the rest of the comments in this section.

We can see that in general, Word2Vec embedding is better in the task of discriminating between related and non related pairs of words for all the considered distances in both filtered and non filtrated schemes. The difference is important up to the point that the best performing distance in GloVe is worse than the worst performing distance in Word2Vec.

Over the results, we can see that for Word2Vec the best performing distances are Cosine and Correlation while the distance we have proposed achieve slightly lower KS (approx. 0.05). In the case of the GloVe filtrated, best options are Cosine, Correlation and Braycurtis, again our proposed distance is slightly inferior (approx. 0.05). In the case of GloVe unfiltered, best options are Canberra and Braycurtis, archiving our proposed distance similar results. Within this results we can see that our proposed distance is robust across embeddings filtered or not, archiving competitive results in all the cases.

Finally, figure 1 compares the values of the KS statistic between synonyms and antonyms distributions for every considered distance in both embeddings. The KS results

[5] `https://github.com/MGijon/Master-s-thesis`

are very small in both embeddings (maximum of 0.12 compare to a range between 0.26 and 0.77 in the previous experiments). This indicates that the antonym relation maintains most of the meaning with minimal seme differences. We can see that in the GloVe embedding the distance is almost the double than in Word2Vec, coherent with the difference of behaviour previously seen.

Table 1. Synonyms: Results for GloVe (dimension 300). The first two columns include the results using all the vocabulary in the embedding while the later two columns use the vocabulary restricted to WN. For each of them we report the KS statistic and the associated p-value

Distance	KS all vocabulary	p-value all vocabulary	KS only WN	p-value only WN
Euclidean	0.2073	$6.0208e^{-170}$	0.4469	0.0
Cosine	0.2623	$1.1153e^{-271}$	0.5368	0.0
Correlation	0.2629	$4.4642e^{-271}$	0.5358	0.0
Canberra	0.4692	0.0	0.4993	0.0
Braycurtis	0.4270	0.0	0.5359	0.0
CS	0.4619	0.0	0.4883	0.0

Table 2. Synonyms: Results for Word2Vec. The first two columns include the results using all the vocabulary in the embedding while the later two columns use the vocabulary restricted to WN. For each of them we report the KS statistic and the associated p-value

Distance	KS all vocabulary	p-value all vocabulary	KS only WN	p-value only WN
Euclidean	0.2926	0.0	0.4153	0.0
Cosine	0.7224	0.0	0.6502	0.0
Correlation	0.7234	0.0	0.6509	0.0
Canberra	0.6287	0.0	0.6170	0.0
Braycurtis	0.7073	0.0	0.6575	0.0
CS	0.6564	0.0	0.6026	0.0

Table 3. Antonyms: Results for GloVe (dimension 300). The first two columns include the results using all the vocabulary in the embedding while the later two columns use the vocabulary restricted to WN. For each of them we report the KS statistic and the associated p-value

Distance	KS all vocabulary	p-value all vocabulary	KS only WN	p-value only WN
Euclidean	0.2692	$2.7072e^{-90}$	0.5286	0.0
Cosine	0.3745	$2.4274e^{-174}$	0.6352	0.0
Correlation	0.3759	$1.2959e^{-175}$	0.6347	0.0
Canberra	0.5801	0.0	0.5954	0.0
Braycurtis	0.5496	0.0	0.6435	0.0
CS	0.5794	0.0	0.5998	0.0

6. Conclusions

1. We provided a methodology to evaluate the quality of a word embedding based on comparing distances between known synonyms and pairs of random words.

Table 4. Antonyms: Results for Word2Vec. The first two columns include the results using all the vocabulary in the embedding while the later two columns use the vocabulary restricted to WN. For each of them we report the KS statistic and the associated p-value

Distance	KS all vocabulary	p-value all vocabulary	KS only WN	p-value only WN
Euclidean	0.3387	$2.3391e^{-155}$	0.4677	$9.8481e^{-296}$
Cosine	0.7700	0.0	0.7035	0.0
Correlation	0.7695	0.0	0.7035	0.0
Canberra	0.6845	0.0	0.6667	0.0
Braycurtis	0.7580	0.0	0.7036	0.0
CS	0.7140	0.0	0.6505	0.0

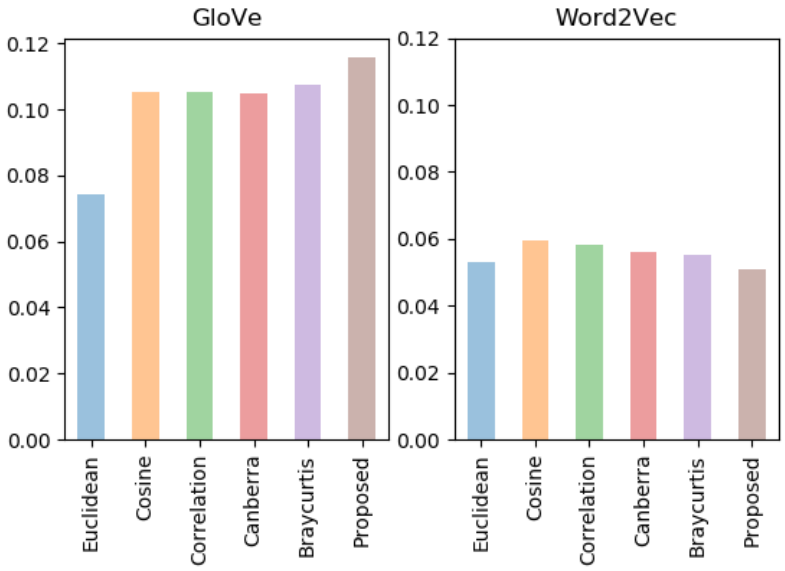

Figure 1. Kolmogorov-Smirnov statistic for the comparison between distances taken from pairs of synonyms and pairs of antonyms in GloVe and Word2Vec scenarios

2. The distributions of distances between synonyms and between antonyms are different but quite similar. It seems not possible to differentiate them in most of the cases based on their embedding distance. We understand that these results are aligned with the intuitive idea that two antonyms have actually a very similar meaning compared to unrelated words.
3. About the norms in the GloVe:
   (a) Vocabulary filtered by WordNet: the most effective distances attend to the norm of the vector that join the two words.
   (b) All the vocabulary: the most effective norms attend to difference between components (Canberra, Braycurtis and the proposed one).

(c) The result that norms attending mostly to component differences perform better would support the hypothesis that if two words are synonyms (or antonyms), they are very similar in almost all components.
(d) The words not present in WordNet behave similarly to unrelated words.

About the norms in the Word2Vec context: there is no big difference between the performances in any context. That indicates that the synonyms and antonyms are related in terms of norm and components and this relation is of a nature such that allows us to distinguish them between other words of the whole vocabulary of the embedding.

4. The proposed norm is better than the others studied in terms of computation time (we did run a series of experiments computing distances between two random vectors and CS distance is aprox. 75% faster than the euclidena distance). This is specially important in problems involving computation of pairwise distance between all embedded instances (*e.g.* clustering, nearest neighbours).
5. Word2Vec embedding performs significantly better than GloVe embedding representing semantically related words (synonyms or antonyms) closer than unrelated. Being this a desirable characteristic of a word embedding, the present work can establish a new criteria for word embedding selection.

7. Future Works

The results obtained are surprising in many aspects, it is intended as future works to train our own embeddings to be able to perform a proper ablation study to identify the root causes.

8. Supplementary Material

The mathematical definition of distance demands that $d(x,y) \geq 0$ and $d(x,y) = 0$ if and only if $x = y$. In the case of the CS distance this does not happen since two different vectors can be at distance 0 (*i.e.* take $x = (1,1,\cdots,1)$ and $y = (2,2,\cdots,2)$, $d_{\text{CS}}(x,y) = 0$). To fix that and make this mathematically rigorous we can define a relation of equivalence like this:

$$u \mathscr{L} v \iff sing(u_i) = sing(v_i) \;\; (\forall i = 1,,N) \tag{9}$$

where the sing function is defined as follows:

$$sing(x) = \begin{cases} 1 & ,x > 0 \\ 0 & ,x = 0 \\ -1 & ,x < 0 \end{cases} \tag{10}$$

This equivalence classes will have this structure:

$$[x] = \{(x_1, ..., x_N) : x_i \in \{-1, 0, 1\}\ (\forall i = 1, ..., N)\}$$

where for a vector $u = (u_1, \cdots, u_N) \in \mathbb{R}^N \longmapsto [x]$, $u \mathscr{L} x$ in this way:

$$x_i = \begin{cases} 1 & , u_i > 0 \\ -1 & , u_i \leq 0 \end{cases}$$

Notice that we are dividing the space in a total of 2^N equivalence classes because the definition of x_i (the components of the equivalence class).

Observation: we can define the distance proposed as follows, for $u \mathscr{L} x,\ v \mathscr{L} y$:

$$d_{\text{proposed}} : \mathbb{R}^N \times \mathbb{R}^N \longrightarrow [0, N] \subset \mathbb{N}$$

$$d_{\text{CS}}(u, v) = d_{\text{CS}}([x], [y]) = \begin{cases} 0 & , \sum_{i=1}^N x_i \cdot y_i \leq 0 \\ \sum_{i=1}^N x_i \cdot y_i & , \text{otherwhise} \end{cases}$$

Now we define the distance proposed not as a distance between vectors, otherwise as a distance between the equivalence classes this vectors belongs to under this relation of equivalence.

Result: distance proposed is a distance between the equivalence classes defined as above.

Proof:

We have to check four properties:

- $d_{\text{CS}}(x, y) \geq 0$, this is true by the definition of norm.
- $d_{\text{CS}}(x, y) = d_{\text{CS}}([x], [y]) = 0 \iff [x] = [y] \iff x \mathscr{L} y$ ie. x and y belongs to the same class, so they are the same in this sense.
- Symmetry: $d_{\text{CS}}(x, y) = d_{\text{CS}}(y, x)$, immediate from the symmetry of the product.
- Triangular inequality: $d_{\text{CS}}(x, z) \leq d_{\text{CS}}(x, y) + d_{\text{CS}}(y, z)$

$$0 \leq \sum_{i=1}^N x_i \cdot z_i \leq \sum_{i=1}^N x_i \cdot y_i + \sum_{i=1}^N y_i \cdot z_i = \sum_{i=1}^N y_i (x_i + z_i)$$

and observe that $x_i, y_i, z_i \in \{-1, 1\}$.

□

References

[1] O. Levy, Y. Goldberg. "Linguistic Regularities in Sparse and Explicit Word Representations" (2014).

[2] G.A. Miller. "WordNet: A Lexical Database for English" (1995).

[3] J. Camacho-Collados, M.T. Pilehvar. "From Word to Sense Embeddings: A Survey on Vector Representations of Meaning" (2018).

[4] P. Jeffrey, R. Socher, and C. Manning. “Glove: Global vectors for word representation”. Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP, 2014).
[5] Y. Goldbert and O. Levy. “Word2Vec Explained: Deriving Mikolov et al.’s Negative-Sampling Word-Embedding Method” (2014).
[6] X. Rong. “Word2Vec Parameter Learning Explained” (2016).
[7] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, J. Dean. “Distributed Representations of Words and Phrases and their Compositionality” (2013).
[8] T. Mikolov, K. Chen, G. Corrado, J. Dean. “Efficient Estimation of Word Representations in Vector Space” (2013).
[9] T. Mikolov, W. Yih, G. Zweig. “Linguistic Regularitues in Continuous Space Word Representations” (2013).
[10] T. Bolukbasi, K.W. Chang, J. Zou, V. Saligrana, A. Kalai. “Man is to Computer Programmer as Woman is to Homemaer? Debiasing Word Embeddings” (2016).
[11] K. Beyer, J. Goldstein, R. Ramakrishnan, U. Shaft. “When Is “Nearest Neighbour” Meaningful?” (1998).
[12] P. Indyk, R. Motwani. “Approximate Nearest Neighbors: Towards Removing the Curse of Dimensionality” (1999).
[13] T.B. Hashimoto, D. Alvarez-Melis, T.S. Jaakkola. “Word Embeddings as Metric Recovery in Semantic Spaces” (2016).
[14] D. Alvarez-Melis, T.S. Jaakkola. “Gromov-Wasserstein Alignment of Word Embedding Spaces” (2018).
[15] D. Garcia-Gasulla, F. Parés, A. Vilalta, J. Moreno, E. Ayguadé, J. Labarta, U. Cortés, T. Suzumura. “On the Behavior of Convolutional Nets for Feature Extraction” (2018).
[16] M. Gijón. “An Analysis of Word Embedding Spaces and Regularities” (2019).